Natural Language Processing

Info 159/259
Lecture 13: Constituency syntax (Oct 4, 2018)

David Bamman, UC Berkeley
Laura McGrath, Stanford

“Corporate Style: The Effect of Comp Titles on Contemporary Literature”

5:30 pm - 7:00 pm (today!)
Geballe Room, Townsend Center, 220 Stephens Hall
Syntax

• With syntax, we’re moving from labels for discrete items — documents (sentiment analysis), tokens (POS tagging, NER) — to the structure between items.

I shot an elephant in my pajamas
I shot an elephant in my pajamas
Why is syntax important?
Why is POS important?

- POS tags are indicative of syntax
- POS = cheap multiword expressions \[(JJ|NN)^+ \text{NN}\]
- POS tags are indicative of pronunciation (“I contest the ticket” vs “I won the contest”)
Why is syntax important?

- Foundation for **semantic analysis** (on many levels of representation: semantic roles, compositional semantics, frame semantics)

http://demo.ark.cs.cmu.edu
Why is syntax important?

- Strong representation for *discourse analysis* (e.g., coreference resolution)

  Bill *VBD* Jon; he was having a good day.

- Many factors contribute to pronominal coreference (including the specific verb above), but syntactic subjects > objects > objects of prepositions are more likely to be antecedents
Why is syntax important?

Linguistic typology; relative positions of subjects (S), objects (O) and verbs (V)

<table>
<thead>
<tr>
<th></th>
<th>Language</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVO</td>
<td>English, Mandarin</td>
<td>I grabbed the chair</td>
</tr>
<tr>
<td>SOV</td>
<td>Latin, Japanese</td>
<td>I the chair grabbed</td>
</tr>
<tr>
<td>VSO</td>
<td>Hawaiian</td>
<td>Grabbed I the chair</td>
</tr>
<tr>
<td>OSV</td>
<td>Yoda</td>
<td>Patience you must have</td>
</tr>
</tbody>
</table>

... ... ...
Sentiment analysis

"Unfortunately I already had this exact picture tattooed on my chest, but this shirt is very useful in colder weather."
[overlook1977]
Barack Hussein Obama II (born August 4, 1961) is the 44th and current President of the United States, and the first African American to hold the office. Born in Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he served as president of the *Harvard Law Review*. He was a community organizer in Chicago before earning his law degree. He worked as a civil rights attorney and taught constitutional law at the University of Chicago Law School between 1992 and 2004.
OBAMA’S DELUSIONAL FOCUS ON GLOBAL WARMING

<table>
<thead>
<tr>
<th>subject</th>
<th>predicate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obama</td>
<td>knows that global warming is a scam.</td>
</tr>
<tr>
<td>Obama</td>
<td>is playing to the democrat base of activists and protesters</td>
</tr>
<tr>
<td>Human activity</td>
<td>is changing the climate</td>
</tr>
<tr>
<td>Global warming</td>
<td>is real</td>
</tr>
</tbody>
</table>
Syntax

• Syntax is fundamentally about the hierarchical structure of language and (in some theories) which sentences are grammatical in a language.

words $\rightarrow$ phrases $\rightarrow$ clauses $\rightarrow$ sentences
Formalisms

Phrase structure grammar
(Chomsky 1957)

Dependency grammar
(Mel’čuk 1988; Tesnière 1959; Pāṇini)
Constituency

- Groups of words ("constituents") behave as single units
- "Behave" = show up in the same distributional environments
<table>
<thead>
<tr>
<th>Context</th>
<th>Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>everyone likes</td>
<td></td>
</tr>
<tr>
<td>a bottle of</td>
<td>is on the table</td>
</tr>
<tr>
<td></td>
<td>makes you drunk</td>
</tr>
<tr>
<td>a cocktail with</td>
<td>and seltzer</td>
</tr>
</tbody>
</table>

from POS 9/25
Parts of speech

• Parts of speech are categories of words defined distributionally by the morphological and syntactic contexts a word appears in.
Syntactic distribution

• Substitution test: if a word is replaced by another word, does the sentence remain grammatical?

<table>
<thead>
<tr>
<th>Kim saw the</th>
<th>elephant</th>
<th>before we did</th>
</tr>
</thead>
<tbody>
<tr>
<td>dog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>idea</td>
<td></td>
<td></td>
</tr>
<tr>
<td>*of</td>
<td></td>
<td></td>
</tr>
<tr>
<td>*goes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bender 2013
Syntactic distributions

<table>
<thead>
<tr>
<th>three parties from Brooklyn</th>
<th>arrive</th>
</tr>
</thead>
<tbody>
<tr>
<td>a high-class spot such as Mindy’s</td>
<td>attracts</td>
</tr>
<tr>
<td>the Broadway coppers</td>
<td>love</td>
</tr>
<tr>
<td>they</td>
<td>sit</td>
</tr>
</tbody>
</table>
Syntactic distributions

Grammatical only when the entire phrase is present, not an individual word in isolation.

<table>
<thead>
<tr>
<th>three parties from Brooklyn</th>
<th>arrive</th>
</tr>
</thead>
<tbody>
<tr>
<td>a high-class spot such as Mindy’s</td>
<td>attracts</td>
</tr>
<tr>
<td>the Broadway coppers</td>
<td>love</td>
</tr>
<tr>
<td>they</td>
<td>sit</td>
</tr>
</tbody>
</table>

Jurafsky and Martin 2017
Syntactic distributions

I’d like to fly from Atlanta to Denver on September seventeenth.
Formalisms

Phrase structure grammar  
(Chomsky 1957)

Dependency grammar  
(Mel’čuk 1988; Tesnière 1959; Pāṇini)
A CFG gives a formal way to define what meaningful constituents are and exactly how a constituent is formed out of other constituents (or words). It defines valid structure in a language.

NP → Det Nominal

NP → Verb Nominal
Context-free grammar

A context-free grammar defines how symbols in a language combine to form valid structures.

<table>
<thead>
<tr>
<th>Non-terminal</th>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP</td>
<td>→ Det Nominal</td>
</tr>
<tr>
<td>NP</td>
<td>→ ProperNoun</td>
</tr>
<tr>
<td>Nominal</td>
<td>→ Noun</td>
</tr>
<tr>
<td>Det</td>
<td>→ a</td>
</tr>
<tr>
<td>Noun</td>
<td>→ flight</td>
</tr>
</tbody>
</table>
Context-free grammar

<table>
<thead>
<tr>
<th>$N$</th>
<th>Finite set of non-terminal symbols</th>
<th>NP, VP, S</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Sigma$</td>
<td>Finite alphabet of terminal symbols</td>
<td>the, dog, a</td>
</tr>
<tr>
<td>$R$</td>
<td>Set of production rules, each $A \rightarrow \beta$</td>
<td>$S \rightarrow$ NP VP</td>
</tr>
<tr>
<td></td>
<td>$\beta \in (\Sigma, N)$</td>
<td>Noun $\rightarrow$ dog</td>
</tr>
<tr>
<td>$S$</td>
<td>Start symbol</td>
<td></td>
</tr>
</tbody>
</table>
Infinite strings with finite productions

Some sentences go on

Bender 2016
Infinite strings with finite productions

• This is the house
• This is the house that Jack built
• This is the cat that lives in the house that Jack built
• This is the dog that chased the cat that lives in the house that Jack built
• This is the flea that bit the dog that chased the cat that lives in the house that Jack built
• This is the virus that infected the flea that bit the dog that chased the cat that lives in the house that Jack built

Smith 2017
Given a CFG, a derivation is the sequence of productions used to generate a string of words (e.g., a sentence), often visualized as a parse tree.
Language

The formal language defined by a CFG is the set of strings derivable from $S$ (start symbol)
The lexicon for \( \mathcal{L}_0 \).

- **Noun** → flights, breeze, trip, morning
- **Verb** → is, prefer, like, need, want, fly
- **Adjective** → cheapest, non-stop, first, latest, other, direct
- **Pronoun** → me, I, you, it
- **Proper-Noun** → Alaska, Baltimore, Los Angeles, Chicago, United, American
- **Determiner** → the, a, an, this, these, that
- **Preposition** → from, to, on, near
- **Conjunction** → and, or, but

The grammar for \( \mathcal{L}_0 \), with example phrases for each rule.

- **Grammar Rules**
  - **S** → NP VP
  - **NP** → Pronoun
    - I
  - **Nominal** → Proper-Noun
    - Los Angeles
  - **Nominal** → Determined Nominal
    - a + flight
  - **Nominal** → Nominal Noun
    - morning + flight
  - **Nominal** → Noun
    - flights
  - **VP** → Verb
    - do
  - **Verb NP** → want + a flight
  - **Verb NP PP** → leave + Boston + in the morning
  - **Verb PP** → leaving + on Thursday
  - **PP** → Preposition NP
    - from + Los Angeles
Bracketed notation

[NP [Det the] [Nominal [Noun flight]]]
Constituents

Every internal node is a phrase

- my pajamas
- in my pajamas
- elephant in my pajamas
- an elephant in my pajamas
- shot an elephant in my pajamas
- I shot an elephant in my pajamas

Each phrase could be replaced by another of the same type of constituent
S → VP

- Imperatives
- “Show me the right way”
S \rightarrow \text{NP VP}

- Declaratives
- “The dog barks”
Yes/no questions

“Will you show me the right way?”

Question generation: subject/aux inversion

“the dog barks” ⇒ “is the dog barking”

S → NP VP ⇒ S → Aux NP VP
S → Wh-NP VP

- Wh-subject-question
- “Which flights serve breakfast?”
Nominal → Nominal PP

- An elephant [PP in my pajamas]
- The cat [PP on the floor] [PP under the table] [PP next to the dog]
Relative clauses

• A relative pronoun (that, which) in a relative clause can be the subject or object of the embedded verb.

• A flight [RelClause that serves breakfast]

• A flight [RelClause that I got]

• Nominal → RelClause

• RelClause → (who | that) VP
## Verb phrases

<table>
<thead>
<tr>
<th>Production</th>
<th>Form</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>VP → Verb</td>
<td></td>
<td>disappear</td>
</tr>
<tr>
<td>VP → Verb NP</td>
<td></td>
<td>prefer a morning flight</td>
</tr>
<tr>
<td>VP → Verb NP PP</td>
<td></td>
<td>prefer a morning flight on Tuesday</td>
</tr>
<tr>
<td>VP → Verb PP</td>
<td></td>
<td>leave on Tuesday</td>
</tr>
<tr>
<td>VP → Verb S</td>
<td></td>
<td>I think [I want a new flight]</td>
</tr>
<tr>
<td>VP → Verb VP</td>
<td></td>
<td>want [VP to fly today]</td>
</tr>
</tbody>
</table>

Not every verb can appear in each of these productions.
## Verb phrases

<table>
<thead>
<tr>
<th>Production</th>
<th>Example</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>VP → Verb</td>
<td>*I filled</td>
<td></td>
</tr>
<tr>
<td>VP → Verb NP</td>
<td>*I exist the morning flight</td>
<td></td>
</tr>
<tr>
<td>VP → Verb NP PP</td>
<td>*I exist the morning flight on Tuesday</td>
<td></td>
</tr>
<tr>
<td>VP → Verb PP</td>
<td>*I filled on Tuesday</td>
<td></td>
</tr>
<tr>
<td>VP → Verb S</td>
<td>*I exist [s I want a new flight]</td>
<td></td>
</tr>
<tr>
<td>VP → Verb VP</td>
<td>* I fill [VP to fly today]</td>
<td></td>
</tr>
</tbody>
</table>

Not every verb can appear in each of these productions.
Subcategorization

• Verbs are compatible with different complements
  • Transitive verbs take direct object NP (“I filled the tank”)
  • Intransitive verbs don’t (“I exist”)
Subcategorization

• The set of possible complements of a verb is its subcategorization frame.

| VP → Verb VP | * I fill [VP to fly today] |
| VP → Verb VP | I want [VP to fly today] |
Coordination

<table>
<thead>
<tr>
<th>Part of Speech</th>
<th>Coordination</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP</td>
<td>$\rightarrow$ NP and NP</td>
<td>the dogs and the cats</td>
</tr>
<tr>
<td>Nominal</td>
<td>$\rightarrow$ Nominal and Nominal</td>
<td>dogs and cats</td>
</tr>
<tr>
<td>VP</td>
<td>$\rightarrow$ VP and VP</td>
<td>I came and saw and conquered</td>
</tr>
<tr>
<td>JJ</td>
<td>$\rightarrow$ JJ and JJ</td>
<td>beautiful and red</td>
</tr>
<tr>
<td>S</td>
<td>$\rightarrow$ S and S</td>
<td>I came and I saw and I conquered</td>
</tr>
</tbody>
</table>

Coordination here also helps us establish whether a group of words forms a constituent.
I shot an elephant in my pajamas
S → NP → VP
   \  /   \
  shot /     \ Nominal
     \       /  PP
      Nominal /   NP
      an     in
     elephant
          /   NP
         my pajamas

S → VP → PP
   \  /   \
  shot /     \ NP
     an   Nominal in my pajamas
          /   NP
         elephant
          /   NP
         my pajamas
Evaluation

Parseval (1991):
Represent each tree as a collection of tuples:

\(<l_1, i_1, j_1>, \ldots, <l_n, i_n, j_n>\)

- \(l_k = \) label for kth phrase
- \(i_k = \) index for first word in kth phrase
- \(j_k = \) index for last word in kth phrase
Evaluation

I_1 shot_2 an_3 elephant_4 in_5 my_6 pajamas_7

- <S, 1, 7>
- <NP, 1, 1>
- <VP, 2, 7>
- <VP, 2, 4>
- <NP, 3, 4>
- <Nominal, 4, 4>
- <PP, 5, 7>
- <NP, 6, 7>
S
   NP
   shot
   NP
   an Nominal
       Nominal PP
           Nominal
           elephant in NP
   my pajamas

S
   NP
   shot NP
   VP
   PP
   in NP
   elephant
   my pajamas
Evaluation

I\textsubscript{1} shot\textsubscript{2} an\textsubscript{3} elephant\textsubscript{4} in\textsubscript{5} my\textsubscript{6} pajamas\textsubscript{7}
Evaluation

Calculate precision, recall, F1 from these collections of tuples

• Precision: number of tuples in tree 1 also in tree 2, divided by number of tuples in tree 1

• Recall: number of tuples in tree 1 also in tree 2, divided by number of tuples in tree 2
Evaluation

I__1__ shot__2__ an__3__ elephant__4__ in__5__ my__6__ pajamas__7__

• \(<S, 1, 7>\)
• \(<NP, 1,1>\)
• \(<VP, 2, 7>\)
• \(<VP, 2, 4>\)
• \(<NP, 3, 4>\)
• \(<Nominal, 4, 4>\)
• \(<PP, 5, 7>\)
• \(<NP, 6, 7>\)
CFGs

- Building a CFG by hand is really hard

- To capture all (and only) grammatical sentences, need to exponentially increase the number of categories (e.g., detailed subcategorization info)

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Verb-with-no-complement</td>
<td>→</td>
<td>disappear</td>
</tr>
<tr>
<td>Verb-with-S-complement</td>
<td>→</td>
<td>said</td>
</tr>
<tr>
<td>VP</td>
<td>→</td>
<td>Verb-with-no-complement</td>
</tr>
<tr>
<td>VP</td>
<td>→</td>
<td>Verb-with-S-complement S</td>
</tr>
</tbody>
</table>
### CFGs

<table>
<thead>
<tr>
<th>Rule</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Verb-with-no-complement</code> → disappear</td>
<td>disappear</td>
</tr>
<tr>
<td><code>Verb-with-S-complement</code> → said</td>
<td>said</td>
</tr>
<tr>
<td><code>VP</code> → <code>Verb-with-no-complement</code></td>
<td></td>
</tr>
<tr>
<td><code>VP</code> → <code>Verb-with-S-complement S</code></td>
<td></td>
</tr>
</tbody>
</table>

- disappear
- said he is going to the airport
- *disappear he is going to the airport
Treebanks

• Rather than create the rules by hand, we can annotate sentences with their syntactic structure and then extract the rules from the annotations.

• Treebanks: collections of sentences annotated with syntactic structure.
Penn Treebank

S
  NP-SBJ
    NP NNP Pierre
    NP NNP Vinken
    ADJP
      NP JJ
        NP CD
          NNS old
        61
        years
    VP
      MD will
        VP
          VB join
          PP-CLR
            IN as
            NP
don
            DT
              NN
                the
                board
            CD
              Nov.
              29
            NNP
              a
              nonexecutive
director
Penn Treebank

Example rules extracted from this single annotation:

<table>
<thead>
<tr>
<th>Rule</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP</td>
<td>NNP NNP</td>
</tr>
<tr>
<td>NP-SBJ</td>
<td>NP, ADJP,</td>
</tr>
<tr>
<td>S</td>
<td>NP-SBJ VP</td>
</tr>
<tr>
<td>VP</td>
<td>VB NP PP-CLR NP-TMP</td>
</tr>
</tbody>
</table>
Penn Treebank

NP → DT JJ NN
NP → DT JJ NNS
NP → DT JJ NN NN
NP → DT JJ JJ NN
NP → DT JJ CD NNS
NP → RB DT JJ NN NN
NP → RB DT JJ JJ NNS
NP → DT JJ JJ NNP NNS
NP → DT NNP NNP NNP NNP JJ NN
NP → DT JJ NNP CC JJ JJ NN NNS
NP → RB DT JJS NN NN SBAR
NP → DT VBG JJ NNP NNP CC NNP
NP → DT JJ NNS , NNS CC NN NNS NN
NP → DT JJ JJ VBG NN NNP NNP FW NNP
NP → NP JJ , JJ ‘‘ SBAR ’’ NNS
CFG

- A basic CFG allows us to check whether a sentence is grammatical in the language it defines.

- Binary decision: a sentence is either in the language (a series of productions yields the words we see) or it is not.

- Where would this be useful?
PCFG

- Probabilistic context-free grammar: each production is also associated with a probability.

- This lets us calculate the probability of a parse for a given sentence; for a given parse tree $T$ for sentence $S$ comprised of $n$ rules from $R$ (each $A \rightarrow \beta$):

$$P(T, S) = \prod_{i}^{n} P(\beta \mid A)$$
# PCFG

<table>
<thead>
<tr>
<th>N</th>
<th>Finite set of non-terminal symbols</th>
<th>NP, VP, S</th>
</tr>
</thead>
<tbody>
<tr>
<td>Σ</td>
<td>Finite alphabet of terminal symbols</td>
<td>the, dog, a</td>
</tr>
<tr>
<td>R</td>
<td>Set of production rules, each</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>A → β [p]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>p = P(β</td>
</tr>
<tr>
<td>S</td>
<td>Start symbol</td>
<td></td>
</tr>
</tbody>
</table>
PCFG

$$\sum_{\beta} P(A \rightarrow \beta) = 1$$

(equivalently)

$$\sum_{\beta} P(\beta \mid A) = 1$$
Estimating PCFGs

How do we calculate $P(A \rightarrow \beta)$?
Estimating PCFGs

\[ \sum_{\beta} P(\beta \mid A) = \frac{C(A \rightarrow \beta)}{\sum_{\gamma} C(A \rightarrow \gamma)} \]

(equivalently)

\[ \sum_{\beta} P(\beta \mid A) = \frac{C(A \rightarrow \beta)}{C(A)} \]
<table>
<thead>
<tr>
<th>A</th>
<th>$\beta$</th>
<th>$P(\beta \mid NP)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP $\rightarrow$ NP PP</td>
<td></td>
<td>0.092</td>
</tr>
<tr>
<td>NP $\rightarrow$ DT NN</td>
<td></td>
<td>0.087</td>
</tr>
<tr>
<td>NP $\rightarrow$ NN</td>
<td></td>
<td>0.047</td>
</tr>
<tr>
<td>NP $\rightarrow$ NNS</td>
<td></td>
<td>0.042</td>
</tr>
<tr>
<td>NP $\rightarrow$ DT JJ NN</td>
<td></td>
<td>0.035</td>
</tr>
<tr>
<td>NP $\rightarrow$ NNP</td>
<td></td>
<td>0.034</td>
</tr>
<tr>
<td>NP $\rightarrow$ NNP NNP</td>
<td></td>
<td>0.029</td>
</tr>
<tr>
<td>NP $\rightarrow$ JJ NNS</td>
<td></td>
<td>0.027</td>
</tr>
<tr>
<td>NP $\rightarrow$ QP -NONE-</td>
<td></td>
<td>0.018</td>
</tr>
<tr>
<td>NP $\rightarrow$ NP SBAR</td>
<td></td>
<td>0.017</td>
</tr>
<tr>
<td>NP $\rightarrow$ NP PP-LOC</td>
<td></td>
<td>0.017</td>
</tr>
<tr>
<td>NP $\rightarrow$ JJ NN</td>
<td></td>
<td>0.015</td>
</tr>
<tr>
<td>NP $\rightarrow$ DT NNS</td>
<td></td>
<td>0.014</td>
</tr>
<tr>
<td>NP $\rightarrow$ CD</td>
<td></td>
<td>0.014</td>
</tr>
<tr>
<td>NP $\rightarrow$ NN NNS</td>
<td></td>
<td>0.013</td>
</tr>
<tr>
<td>NP $\rightarrow$ DT NN NN NN</td>
<td></td>
<td>0.013</td>
</tr>
<tr>
<td>NP $\rightarrow$ NP CC NP</td>
<td></td>
<td>0.013</td>
</tr>
</tbody>
</table>
PCFGs

- A CFG tells us whether a sentence is in the language it defines
- A PCFG gives us a mechanism for assigning scores (here, probabilities) to different parses for the same sentence.
$P(\text{NP VP } | \text{ S})$
\[ P(\text{NP VP} | \ S) \times P(\text{Nominal} | \ \text{NP}) \]
\[ P(\text{NP VP | S}) \times P(\text{Nominal | NP}) \times P(\text{Pronoun | Nominal}) \]
\[
P(NP \ VP | S) \\
\times P(\text{Nominal} | NP) \\
\times P(\text{Pronoun} | \text{Nominal}) \\
\times P(I | \text{Pronoun})
\]
\[
P(NP \ VP | S) \\
\times P(Nominal | NP) \\
\times P(Pronoun | Nominal) \\
\times P(I | Pronoun) \\
\times P(VP PP | VP)
\]
\begin{align*}
P(NP \ VP | S) \\
\times P(\text{Nominal} | NP) \\
\times P(\text{Pronoun} | \text{Nominal}) \\
\times P(I | \text{Pronoun}) \\
\times P(\text{VP PP} | \text{VP}) \\
\times P(\text{Verb NP} | \text{VP})
\end{align*}
S
  /\   \\
 NP   VP
  /\   \\
 Nominal  Verb
       /\   |
      NP   Det
           |
           Nominal

P(NP VP | S)
×P(Nominal | NP)
×P(Pronoun | Nominal)
×P(I | Pronoun)
×P(VP PP | VP)
×P(Verb NP | VP)
×P(shot | Verb)
×P(Det Nominal | NP)
×P(an | Det)
P(NP VP | S) 
\times P(Nominal | NP) 
\times P(Pronoun | Nominal) 
\times P(I | Pronoun) 
\times P(VP PP | VP) 
\times P(Verb NP | VP) 
\times P(shot | Verb) 
\times P(Det Nominal | NP) 
\times P(an | Det) 
\times P(Noun | Nominal)
P(elephant | Noun)

P(Verb NP | VP)

P(VP PP | VP)

P(I | Pronoun)

P(Nominal | NP)

P(Pronoun | Nominal)

P(NP VP | S)

P(Det Nominal | NP)

P(Noun | Nominal)

P(an | Det)

P(shot | Verb)

P(an | Noun)

P(elephant | Noun)
\[ P(T, S) = \prod_{i} P(\beta | A) \]
PCFGs

• A PCFG gives us a mechanism for assigning scores (here, probabilities) to different parses for the same sentence.

• But we often care about is finding the single best parse with the highest probability.
Tuesday

• Guest lecture (David Gaddy) on context-free parsing algorithms (will show up on midterm).

• Read (carefully!) chs. 11 and 12 in SLP3, esp re: CKY.